

An Event-Driven Manufacturing Information System Architecture

Alfred Theorin * Kristofer Bengtsson ** Julien Provost ***
Michael Lieder **** Charlotta Johnsson *
Thomas Lundholm **** Bengt Lennartson †

* Lund University, Lund, Sweden

(alfred.theorin@control.lth.se, charlotta.johnsson@control.lth.se).

** Sekvens AB, Gothenburg, Sweden (kristofer@sekvens.se).

*** Technische Universität München, Munich, Germany
(provost@ses.mw.tum.de).

**** KTH Royal Institute of Technology, Stockholm, Sweden
(lieder@kth.se, thomas.lundholm@ip.kth.se).

† Chalmers University of Technology, Gothenburg, Sweden
(bengt.lennartson@chalmers.se).

Abstract: Future manufacturing systems need to be more flexible, to embrace tougher and constantly changing market demands. They also need to make better use of plant data, ideally utilizing all data from the entire plant. Low-level data should be refined to real-time information for decision making, to facilitate competitiveness through informed and timely decisions.

The *Line Information System Architecture*, LISA, is designed to enable flexible factory integration and data utilization. In LISA, international standards and established off-the-shelf technologies have been combined with the main objective to be industrially applicable. LISA is an event-driven architecture with a prototype-oriented information model and formalized transformation services. It features loose coupling, flexibility, and ease of retrofitting legacy devices. The architecture has been evaluated on both real industrial data and industrial demonstrators and is also being installed at a large automotive company.

© 2015, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: industry automation, agile manufacturing, flexible manufacturing systems, architectures, events, decision support systems, automobile industry

1. INTRODUCTION

A hidden asset in manufacturing industry is data. Investigations estimate that 85% of the data are unstructured, and 42% of all transactions (sending and receiving information) are paper-based (IBM (2007)). CEOs in the manufacturing industry say that “we need to do a better job to capture and understand information rapidly in order to make sound business decisions” (Hill and Smith (2009)).

Future industrial manufacturing systems need to make better use of the data (Panetto and Molina (2008)). The low-level data has to be transformed into information that can be used for decision making. In addition, future manufacturing systems need to be productive, flexible, competitive, sustainable, secure, and safe and must reduce waste of material, capital, energy, and media. Most automotive companies use advanced information systems (Dai et al. (2012)). However, most of these systems lack many key features. Improved control, optimization, and human interaction in manufacturing processes is also important for future manufacturing (Blanc et al. (2008)).

There are several types of manufacturing information systems, such as Manufacturing Execution Systems (MES) (Dai et al. (2012)), Enterprise Resource Planning (ERP) (Umble et al. (2003)), or Multi-Agent Systems (MAS)

(Leitao et al. (2013)). These systems require information about the real-time performance and behavior of the manufacturing plant. However, there is no vendor-independent integration architecture for such information management and many companies have developed their own solutions. With an increasing demand to launch new vehicle models faster, automotive companies require flexible and scalable information systems.

To enable access to the data, all devices and software must first be integrated. To accomplish this, a flexible architecture is needed which facilitates integration of any application or device. Plants often use a wide range of devices, based on different technologies from different eras. Some devices originate from when the plant was built, and devices have then been added as part of continuous improvements. Retrofitting legacy devices is thus a particularly important aspect. It must be possible to integrate them regardless of their capabilities or technology.

The contribution of this paper is a new information system architecture, called Line Information System Architecture (LISA), that enables flexibility and scalability. The architecture is event-based, has formalized transformation patterns, and uses stream-based aggregation and prototype-oriented information models. LISA is able to handle layout and structural changes on the plant floor and allows a

large diversity of devices and applications. Furthermore, LISA enables new Key Performance Indicators (KPIs) to be calculated, not only for new, but also for historical data. LISA has been implemented and evaluated on industrial data and demonstrators, and is being installed at a large automotive company.

In Section 2, the concepts of service-oriented and event-driven architectures are introduced. In Section 3, LISA is described. How LISA can be used for KPI calculation is explained in Section 4. Finally, event-based control with LISA is presented in Section 5.

2. SERVICE-ORIENTED ARCHITECTURES

When new functionality and systems are added, they need to be rapidly integrated with existing systems. The traditional integration approach in manufacturing is to connect applications on a Point-to-Point (PtP) basis using the client/server pattern. The pattern mandates that the server and the client know about each other. The number of connections in a fully connected network increases quadratically with the number of applications. This is known as “spaghetti integration” and makes the system rigid and hard to maintain (Boyd et al. (2008)). Each time an application is added, all other applications need to be updated to be able to interact with the new application.

It is common that applications can only communicate through proprietary or specific protocols, and applications may require external message translators to communicate with each other. This is, for example, the normal case for communication between Programmable Logic Controllers (PLCs) from different vendors. Another challenge is communication between the different levels of ISA95, see Fig. 1, known as vertical integration.

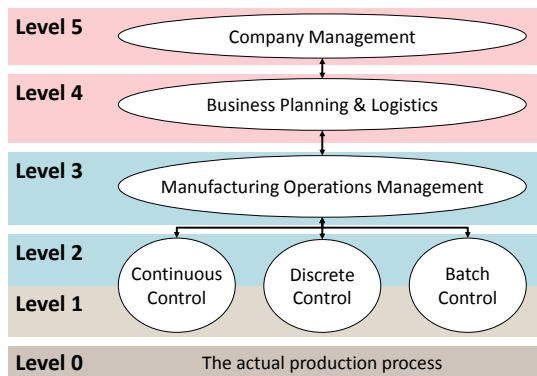


Fig. 1. Functional hierarchy as defined by ISA95.

The PtP approach poorly supports business requirements (Ribeiro et al. (2008)). Yet, industry has been slow to migrate to new approaches, mainly due to the cost of replacing their established legacy systems based on PtP (Boyd et al. (2008)). However, migration has been significantly accelerated by the advent of Service-Oriented Architectures (SOAs) (He and Xu (2014)).

2.1 Service-Oriented Architecture

SOA is a distributed software architecture where self-contained applications expose themselves as services,

which other applications can connect to and use. To reach its full potential, SOA applications should be self-describing, discoverable, and platform- and language-independent. This leads to loose coupling and high flexibility.

SOA has recently received much attention in both academia and industry. The adoption of SOA in a company typically starts as an IT initiative to improve infrastructure efficiency and can then mature into optimized use for business purposes (Welke et al. (2011)). SOA is widely used on the business level and is expected to revolutionize manufacturing in a similar fashion.

The further down the hierarchy in Fig. 1, the shorter the task time frame. On level 1 it is common with hard real-time requirements, with deadlines in the order of milliseconds. The devices which execute on level 1 often have strictly limited memory and computational power. There is a trade-off between flexibility and real-time performance (Theiss et al. (2009)) and thus, the further down SOA is wanted, the more performant (and hence less flexible) it needs to be. Most SOA tools are tailored for business processes, which do not have strict timing or resource requirements. Thus, these tools cannot be used for manufacturing processes. However, there have been initiatives to bring SOA to level 1 and 2 by customizing the web service technology for resource constrained devices (Cucinotta et al. (2009); Dai et al. (2014)).

2.2 Event-Driven Architecture

Even though SOA conceptually offers loose coupling and is intended to be distributed, service orchestration is typically done centrally, with the orchestrator taking control of the involved services. SOA 2.0, also known as advanced SOA or event-driven SOA, is the next generation of SOA that focuses on events, inspired by Event-Driven Architecture (EDA). SOA 2.0 enables service choreography, where each service reacts to published events on its own, rather than being requested to do so by a central orchestrator.

EDA is extremely loosely coupled and highly distributed by design. An event creator only needs to know that the event occurred, it does not need to know anything about who is interested in the event or how it will be processed (Michelson (2006)). Event data should be immutable since it is then always (thread-)safe to send the events within and between applications. With EDA, applications turn from synchronized and blocking to asynchronous and non-blocking (Kuhn and Allen (2014)).

3. LINE INFORMATION SYSTEM ARCHITECTURE

LISA is an EDA that provides loose coupling of applications and devices, as well as a flexible message structure for integration. The core components of LISA are the message bus, the LISA message format, and communication and service endpoints. They enable creation and transformation of events into usable information in a loosely coupled way, and will be described in the following sections.

3.1 LISA Events

A common approach for information systems is an object-oriented structure for event types and events (Cheng et al.

Download English Version:

<https://daneshyari.com/en/article/711965>

Download Persian Version:

<https://daneshyari.com/article/711965>

[Daneshyari.com](https://daneshyari.com)